APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/823,157	04/13/2004	Christopher Martin Lott	APP 1556	4543

			EXAMINER
9941	7590	11/02/2005	PATEL, MANGLESH M

TELCORDIA TECHNOLOGIES, INC.
ONE TELCORDIA DRIVE 5G116
PISCATAWAY, NJ 08854-4157

ART UNIT	PAPER NUMBER
2178	

DATE MAILED: 11/02/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	10/823,157	LOTT ET AL.
	Examiner	Art Unit
	Manglesh M. Patel	2178

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 April 2004*.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-32* is/are pending in the application.

   4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-32* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *13 April 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   a)☐ All   b)☐ Some * c)☐ None of:

   1.☐ Certified copies of the priority documents have been received.

   2.☐ Certified copies of the priority documents have been received in Application No. _____.

   3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

   * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
   Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to communications: Application filed on April 13, 2004.

2.   Claims 1-32 are pending. Claims 1 and 19 are independent claims.

3.   Acknowledgement is made to applicant's claim for priority to U.S. Application

Serial No. 60/502443, filed on September 12, 2003.

Drawings

4.   The examiner has accepted the Drawings filed on April 13, 2004.

Specification

5.   Applicant is reminded of the proper language and format for an abstract of the
disclosure.

        The abstract should be in narrative form and generally limited to a single
paragraph on a separate sheet within the range of 50 to 150 words. It is important that
the abstract not exceed 150 words in length since the space provided for the abstract
on the computer tape used by the printer is limited. The form and legal phraseology
often used in patent claims, such as "means" and "said," should be avoided. The
abstract should describe the disclosure sufficiently to assist readers in deciding whether
there is a need for consulting the full patent text for details.

        The language should be clear and concise and should not repeat information
given in the title. It should avoid using phrases which can be implied, such as, "The
disclosure concerns," "The disclosure defined by this invention," "The disclosure
describes," etc.

The specification is objected under 37 CFR 1.72(b) because the abstract exceeds 150

words.

Claim Rejections - 35 USC § 102

6.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

7.   Claims 1-7, 9-17, 19-26, 30 & 32 are rejected under 35 U.S.C. 102(e) as being

anticipated by Yehia (U.S. Pub 2002/0147726, filed March 20, 2002).

Regarding Independent claim 1, Yehia discloses *a method for the automatic*

generation of message validation and/or transformation software from message

interface specifications and business rules for use in a message processing

system comprising the steps of:

-   *Inputting a set of message definitions, data dictionary entries and/or*

    business rules using a structured editor to create a set of structured files

    that define the message interface and/or business rules (paragraph 23 &

    85, wherein an interface allows input for rules for the data and rules

    analysis engine that include business rules. The interface guides the user

    in defining the rules and rule data. The compilation component compiles

    the rules and rule data into an XML XSL based rule language therefore

providing a set of structured files that define the message interface or

business rules);

- *Generating message validation software from the set of structured files*

  (paragraphs 270, 272 & 279, wherein client validation and enforcing

  business rules at front-end applications are performed from the set of

  structured files represented within XML documents. The structured xml

  files are used to perform rule validation);

- *Storing the message validation and transformation software in one or*

  more databases for use by the message processing system (paragraph

  279, wherein the client validation creation and distribution is stored within

  a database).

Regarding Dependent claim 2, Yehia discloses *wherein the set of message*

definitions and data dictionary entries can be reused to develop additional

message definitions, dictionary entries and/or business rules across an

enterprise (paragraph 92, wherein the use of XML enables maximal reuse of

information and data through the composition of XML fragments. The message

definitions and data dictionary entries defined within the structured files are

reused because they are described using XML).

Regarding Dependent claim 3, Yehia discloses *further comprising the step of*

generating message transformation software from the set of structured files

(paragraph 94, wherein the action processor is used with the analysis engine for processing actions pertaining to messages. Software components are developed to address the received actions and include structured files since they are described using XML).

Regarding Dependent claim 4, Yehia discloses *wherein the structured files are in the XML format* (paragraph 22, wherein the invention uses standard XML notations to define rules and standard XSL and XSLT processing instructions to enforce rules. Therefore the structured files used to represent the business rules are described using XML).

Regarding Dependent claim 5, Yehia discloses *wherein the generation of message transformation software and message validation software further comprises the step of translating the structured files in the XML format into Extensible Stylesheet Language Transforms (XSLT)* (paragraph 22, wherein the invention uses standard XML notations to define rules and standard XSL and XSLT processing instructions to enforce rules. Therefore the structured XML files are translated into XSLT).

Regarding Dependent claim 6, Yehia discloses *wherein the step of generating message validation software further comprises the step of inputting the structured files into a schema generator in order to generate a set of W3C XML*

Schema to be used to validate messages (paragraphs 97,101 & 195 wherein the

schema details the relationship between members and objects. The schema

shows the relation between orders and services being ordered. The same

relation is carried through the XML fragments. In addition the XML document

values are mapped into the schema for validation).

Regarding Dependent claim 7, Yehia discloses *wherein the step of inputting*

further comprises the step of validating the structured files to ensure the

structured files conform to a pre-determined structure (paragraph 23, wherein the

GUI guides the user in defining the rules and rule data, therefore the structured

files must conform to a pre-determined structure).

Regarding Dependent claim 9, Yehia discloses *wherein the structured files are*

presented to the user in HTML (paragraph 138, wherein one control produces

satisfactory results is scripts embedded in HTML pages. The control is used to

help facilitate the negotiation process, therefore the structured files are presented

to the user in HTML format to display and help guide the negotiation process).

Regarding Dependent claim 10, Yehia discloses *wherein the structured rule*

editor uses a web browser to present the HTML to the user (paragraph 186,

wherein the client connector includes only a browser based GUI that

communicates directly with a web server at the hub. The GUI representing the

structured rules editor uses the web browser to communicate thereby presenting

the HTML to the user).

Regarding Dependent claim 11, Yehia discloses *producing a report detailing*

differences between two sets of structured files (paragraph 130, wherein the

invention checks all related contracts, verifies and analyzes the effect and alerts

the member about any potential conflict. Therefore it is inherent that it identifies

any differences between the structured files since it verifies and alerts the user if

differences exist between the structured files representing the contracts).

Regarding Dependent claim 12, Yehia discloses *modifying the structured files*

after generating the XML schema in order to correct errors identified by the

schema generator (paragraph 196-198 & 230, wherein a rule entity schema is

used with an analysis engine to retrieve rules, rule parameters, actions and

pointers. Therefore the GUI allows the user to update any errors by making

changes to the rules once the rule entity schema indicates errors. A schema

template is used to save the updated information).

Regarding Dependent claim 13, Yehia discloses *inputting the set of interface*

schemas into a schema validator to determine if the generated schemas are

correctly formatted and consistent (paragraphs 101 & 103, wherein the rules

analyzer determines if the generated schema from the database are correctly

formatted and consistent).

Regarding Dependent claim 14, Yehia discloses *modifying the structured files*

after validating the schema in order to correct errors identified by the schema

validator (paragraph 270, wherein the rule handler is used to validate the

structured files from the schema database).

Regarding Dependent claim 15, Yehia discloses *transferring pre-existing word*

processing formatted business rule documents into structured files (paragraph

145, wherein templates that include formatted business rules within word

documents are supported by the invention).

Regarding Dependent claim 16, Yehia discloses *importing from existing W 3 C*

XML Schema files into a set of structured files (paragraphs 92,101 &118, wherein

the templates are built using XML once the data elements are extracted from the

parser. The database used to store them represents the XML schema files and

they are used for building contract templates according to the W3C standard).

Regarding Dependent claim 17, Yehia discloses

- *Translating the word processing formatted document into an XML*

  formatted document (paragraphs 145 & 146, wherein the word document

with headers specifying the contract clause titles is translated into XML
format using XSL);

- *Parsing the XML formatted document to identify unparseable constructs
  and errors* (paragraphs 148 & 197, wherein the parser is used to retrieve
  the contents of specific XML tags, those that describe the metadata of the
  contract. Therefore the XML document is parsed to identify unparseable
  constructs and errors by using the data and rules analysis engine);

- *Presenting the unparseable constructs and errors to the requirements
  engineer for modification* (paragraph 99, wherein the parser is used in
  conjunction with the rules analysis engine, therefore if unparseable
  constructs and errors exist they are modified via GUI that modify the rules
  engine);

- *Rewriting the unparseable constructs into a structured construct using the
  structured rule editor* (paragraphs 92 & 93, wherein the unparseable
  constructs identified by the rules analysis engine are modified using the
  GUI or structured rule editor);

- *And, repeating the parsing, presenting and rewriting steps until all
  unparseable constructs and errors are substantially eliminated*
  (paragraphs 92 & 93, wherein once the rules are modified within the GUI
  they are parsed again using the structured document template).

Regarding Independent claim 19, Yehia discloses *a system for the automatic*

generation of message validation software and/or transformation software from

business rules for use in a message processing system comprising:

- *A structured editor for inputting a set of message definitions, data*

  dictionary entries and business rules to form a set of structured files that

  defines the message interface as a set of nested elements and groups of

  elements and business rules (paragraph 23 & 85, wherein an interface

  allows input for rules for the data and rules analysis engine that include

  business rules. The interface guides the user in defining the rules and rule

  data. The compilation component compiles the rules and rule data into an

  XML XSL based rule language therefore providing a set of structured files

  that define the message interface or business rules);

- *Means for generating message validation software from the structured*

  files (paragraphs 270, 272 & 279, wherein client validation and enforcing

  business rules at front-end applications are performed from the set of

  structured files represented within XML documents. The structured xml

  files are used to perform rule validation);

- *Storage means for storing the message validation and transformation*

  software in one or more databases for use by the message processing

  system (paragraph 279, wherein the client validation creation and

  distribution is stored within a database).

Regarding Dependent claim 20, Yehia discloses *means for generating*

message transformation software from the structured files (paragraph 94,

wherein the action processor is used with the analysis engine for processing

actions pertaining to messages. Software components are developed to address

the received actions and include structured files since they are described using

XML).

Regarding Dependent claim 21, the claim is for a computer system performing

the method of claim 2, and is similarly rejected under the same rationale.

Regarding Dependent claim 22, Yehia discloses *wherein the means for*

generating the schema inputs to message validation software from the structured

files is an XML schema generator (paragraphs 97,101 & 195 wherein the schema

details the relationship between members and objects. The schema shows the

relation between orders and services being ordered. The same relation is carried

through the XML fragments. In addition the XML document values are mapped

into the schema for validation).

Regarding Dependent claim 23, the claim is for a computer system performing

the method of claim 5, and is similarly rejected under the same rationale.

Regarding Dependent claim 24, Yehia discloses *wherein the structured editor comprises a means for constraining the inputs into the structured files to ensure the structured files conform to a pre-determined structure and content* (paragraph 23, wherein the GUI guides the user in defining the rules and rule data, therefore the structured files must conform to a pre-determined structure and content).

Regarding Dependent claim 25, Yehia discloses *wherein the structured editor comprises a graphical user interface for editing the structured files in a tabular non-XML format* (paragraph 23, wherein a structured editor represented by a graphical user interface is used for editing the structured files that represent templates and include documents in a tabular fashion).

Regarding Dependent claim 26, Yehia discloses *wherein the structured editor limits the selection of attributes available to a user during definition of an element, group of elements or rule* (paragraph 23, wherein the framework determines the absence of an attribute, retrieves the rule definition, applies the rule handler, and returns the success or failure codes).

Regarding Dependent claim 30, Yehia discloses *wherein the structured editor is a table editor which enables the user to input tables selected from the group consisting of: Message Definition Tables, Data Dictionary Tables, Business Rule Tables, Error Tables, Variable Definition Tables, and Requirements Trace Matrix*

Tables (paragraph 23, it is inherent that the structured editor is capable of supporting table inputs since it guides the user in defining the rules which are then stored within a schema database in a tabular format).

Regarding Dependent claim 32, Yehia discloses *compare tool for comparing a first structured file or document with a second structured file or document in order to develop a list of differences between such files or documents* (paragraph 130, wherein the invention checks all related contracts, verifies and analyzes the effect and alerts the member about any potential conflict. Therefore it is inherent that it identifies any differences between the structured files since it verifies and alerts the user if differences exist between the structured files representing the contracts, The member is alerted when differences between the contracts exist, therefore a list showing the differences is inherently present).

Claim Rejections - 35 USC § 103

8.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9.   Claims 8, 18, 29 & 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yehia (U.S. Pub 2002/0147726, filed March 20, 2002) in view of Abrari (U.S. Pub 2002/0120917, filed Nov 26, 2001).

Regarding Dependent claim 8, Yehia teaches the creation and representation of business rule definitions using standard XML notation. The creation of the business rules is accomplished using a graphical user interface with a compilation component. The interface guides the user for defining the rules and data associated with the rules. However Yehia fails to explicitly teach an editor for editing structured files in a tabular format (paragraphs 21, 22 & 23). Abrari teaches *wherein the structured editor provides a graphical user interface to the requirements engineer for editing the structured files in a tabular format* (figures 6-17, wherein a tabular format is shown within the graphical user interface for editing the structured files). Yehia and Abrari are analogous art because they are from the same field of endeavor of development of Business rule applications. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include a tabular format within a GUI for editing structured files. The motivation for doing so would have been to allow easier updating and manipulation of programmed code by making changes within a GUI showing the code in a tabular format. Therefore it would have been obvious to combine the teachings of Abrari with Yehia for the benefits of allowing flexible code manipulation by changing structured files using a GUI representing code in a tabular format.

Regarding Dependent claim 18, Yehia fails to explicitly teach the generation of test cases for testing the business rule definitions used for message validation

(paragraphs 21, 22 & 23). Abrari teaches *generating a set of test cases to provide test messages with which to test the message transformation and message validation software* (paragraph 11,47 & 42 wherein the platform separates business rules from procedural business process logic and thereby improving code quality and reducing development costs. In addition the platform enables non-technical personnel to develop, **test**, deploy and update business rules without programming. Therefore it is inherent that testing includes generating test cases to test the message validation software). Yehia and Abrari are analogous art because they are from the same field of endeavor of development of Business rule applications. At the time of the invention it would have been obvious to a person of ordinary skill in the art to include the testing of the message validation software by preparing test cases. The motivation for doing so would have been to minimize errors thereby improving code quality by testing the business rules using test messages. Therefore it would have been obvious to combine the teachings of Abrari with Yehia for the benefits of allowing testing of messages for improving code quality thereby reducing development and maintenance costs.

Regarding Dependent claim 29, Yehia fails to teach a project interface to access all the structured files. Abrari teaches *a project interface for providing access to the user of all structured files used to define the project and access to all functions that can be performed on such files* (paragraph 42, 43 & 46 wherein

building and testing involves the entire project cycle. In addition business rules

developed early in the development process are incorporated into the final

application. It is inherently present that the user has access to all components for

the development of the project including attributes, building, testing and user

interface development. This is represented by the tree view). Yehia and Abrari

are analogous art because they are from the same field of endeavor of

development of Business rule applications. At the time of the invention it would

have been obvious to a person of ordinary skill in the art to allow a user to have

access to all the structured files for project development. The motivation for doing

so would have been to lower personnel costs and increase profits by providing

an integrated and dynamic platform for application development. Therefore it

would have been obvious to combine the teachings of Abrari with Yehia for the

benefits of allowing an integrated set of tools for the development, deployment

and maintenance of applications within a platform.

Regarding Dependent claim 31, Yehia fails to teach a document generator to

develop user-readable documentation pertaining to the message definition

interface and business rules. Abrari teaches *a document generator for generating*

user-readable documentation specifying the message definition interface and

business rules (paragraph 42, wherein the methodology guarantees that UML-

conforming documentation such as case models will be created as an inherent

byproduct of the software development process. Therefore a document is

generated specifying the message definition interface and business rules since

they are part of the application represented by the document). Yehia and Abrari

are analogous art because they are from the same field of endeavor of

development of Business rule applications. At the time of the invention it would

have been obvious to a person of ordinary skill in the art to include

documentation specifying the message definitions and business rules. The

motivation for doing so would have been to provide a more updated platform with

business rules. Therefore it would have been obvious to combine the teachings

of Abrari with Yehia for the benefits of allowing a dynamic platform capable of

generating documentation specifying business rules and message definition

interface.

10.   Claims 27 & 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Yehia (U.S. Pub 2002/0147726, filed March 20, 2002) in view of Tomm (U.S. 6,560,608,

filed Jun 9, 2000).

Regarding Dependent claim 27, Yehia teaches the creation and

representation of business rule definitions using standard XML notation. The

creation of the business rules is accomplished using a graphical user interface

with a compilation component. The interface guides the user for defining the

rules and data associated with the rules. However Yehia fails to explicitly teach

the generation of an index listing of elements used in a definition (paragraphs 21,

22 & 23). Tomm teaches *means for generating an index listing of all elements*

used in an interface definition, cross referencing entries within data dictionaries

with their appearances within message definitions (See figure 4, column 4, lines

23-50, wherein synonyms dictionary has corresponding set of synonyms where

each synonym corresponds to a data field for a given format. Therefore the index

listing is part of the synonym dictionary since it relates to a data field). At the time

of the invention it would have been obvious to a person of ordinary skill in the art

to include an index listing of elements. The motivation for doing so would have

been to provide mapping of data dictionaries by accessing an index listing

thereby allowing rules to be more easily manipulated. Therefore it would have

been obvious to combine the teachings of Abrari with Tomm for the benefits of

eliminating the error-prone and tiresome process of entering the procedure

manually each time it is needed by providing an index listing used to map

dictionary data.

Regarding Dependent claim 28, Yehia fails to teach the use of a data dictionary

capable of providing changes pertaining to only the interface definitions. Tomm

teaches *means for pruning a data dictionary into a data dictionary that comprises*

only those elements and/or group of elements that are used in a message

interface definition (column 5, lines 25-42, wherein the rules dictionary is

searched for rules matching the signature, if no rule matches an editor allows the

user to create a rule. The rules or elements are defined using the editor which

inherently includes removal of definitions from the dictionary or modifying the

dictionary to only show elements pertaining to the message interface definition).

At the time of the invention it would have been obvious to a person of ordinary

skill in the art to include the removal/manipulation of rules within a data

dictionary. The motivation for doing so would have been to provide rules

adaptable to a target message format, thereby improving data mapping providing

easier data manipulation. Therefore it would have been obvious to combine the

teachings of Abrari with Tomm for the benefits of providing an editor to make

changes within a data dictionary for mapping target messages, thereby providing

easier data manipulation and conversion between messages.

Other Prior Art Cited

11.   The prior art made of record and not relied upon is considered pertinent to

applicant's disclosure.

- Serrano-Morales et al. (U.S. Pub 2002/0032688) discloses "Approach For
  Re-Using Business Rules"

Conclusion

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Manglesh M. Patel whose telephone number is (571)

272-5937. The examiner can normally be reached on M,F 8:30-6:00 T,TH 8:30-3:00
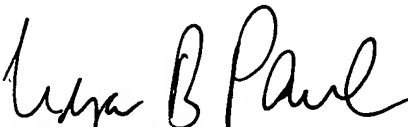
Wed 8:30-7:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Stephen S. Hong can be reached on (571)272-4124. The fax phone

number for the organization where this application or proceeding is assigned is 571-

273-8300.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

Manglesh M. Patel

Patent Examiner

October 25, 2005

CESAR PAULA

PRIMARY EXAMINER